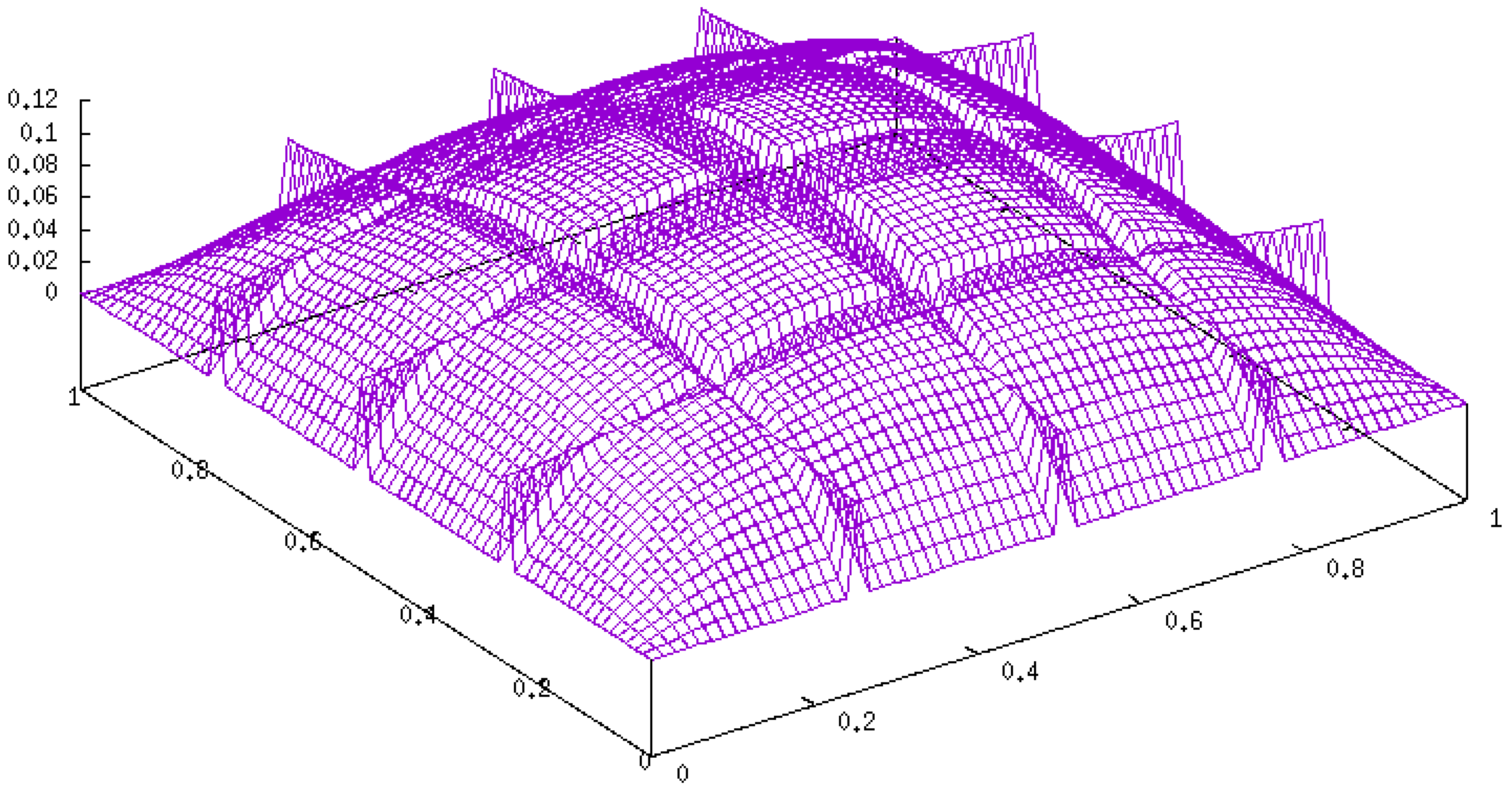
MPI Parallel Multigrid
2D Laplace/Poisson Eq.

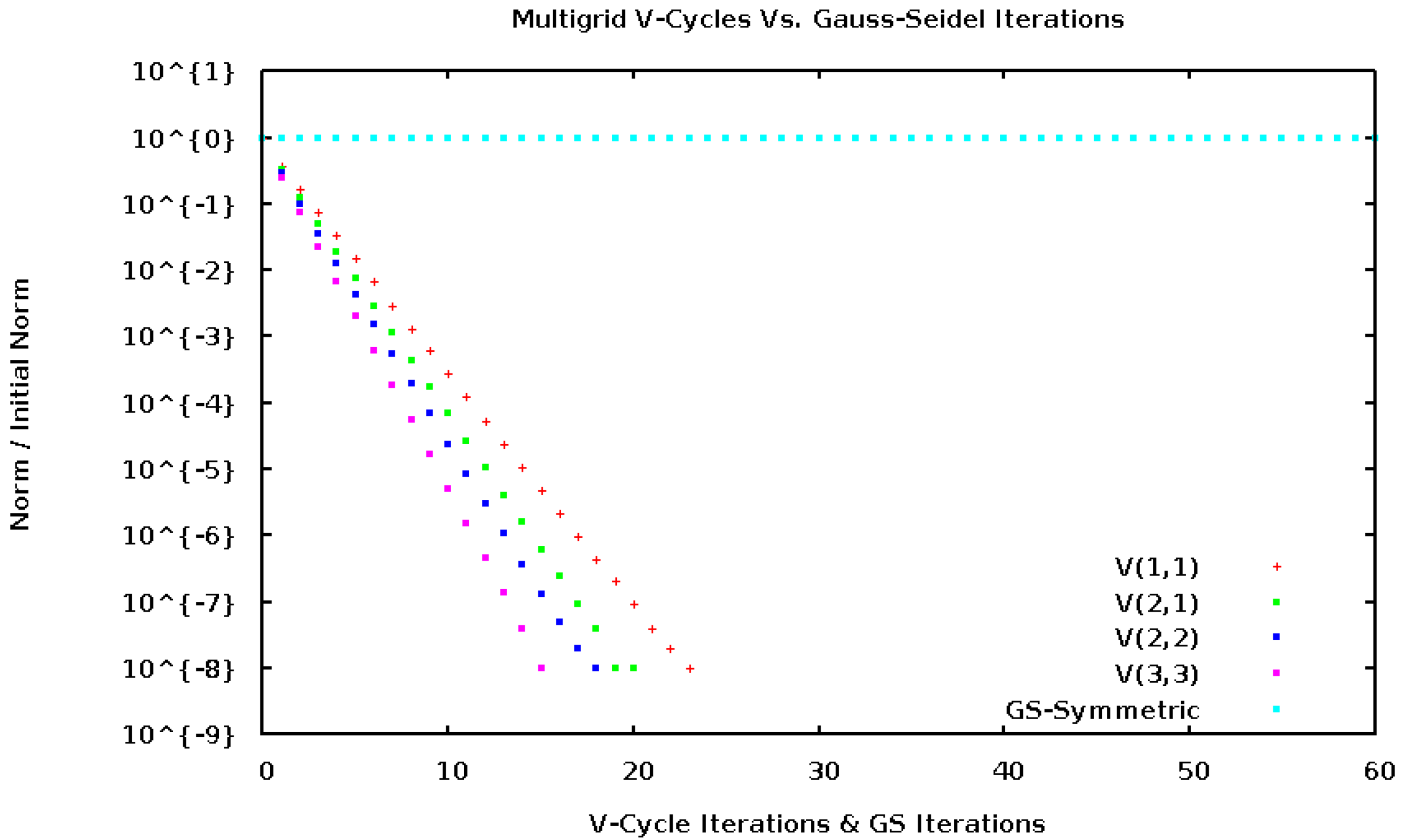
Joe Zuhusky
MSCS



Problem:

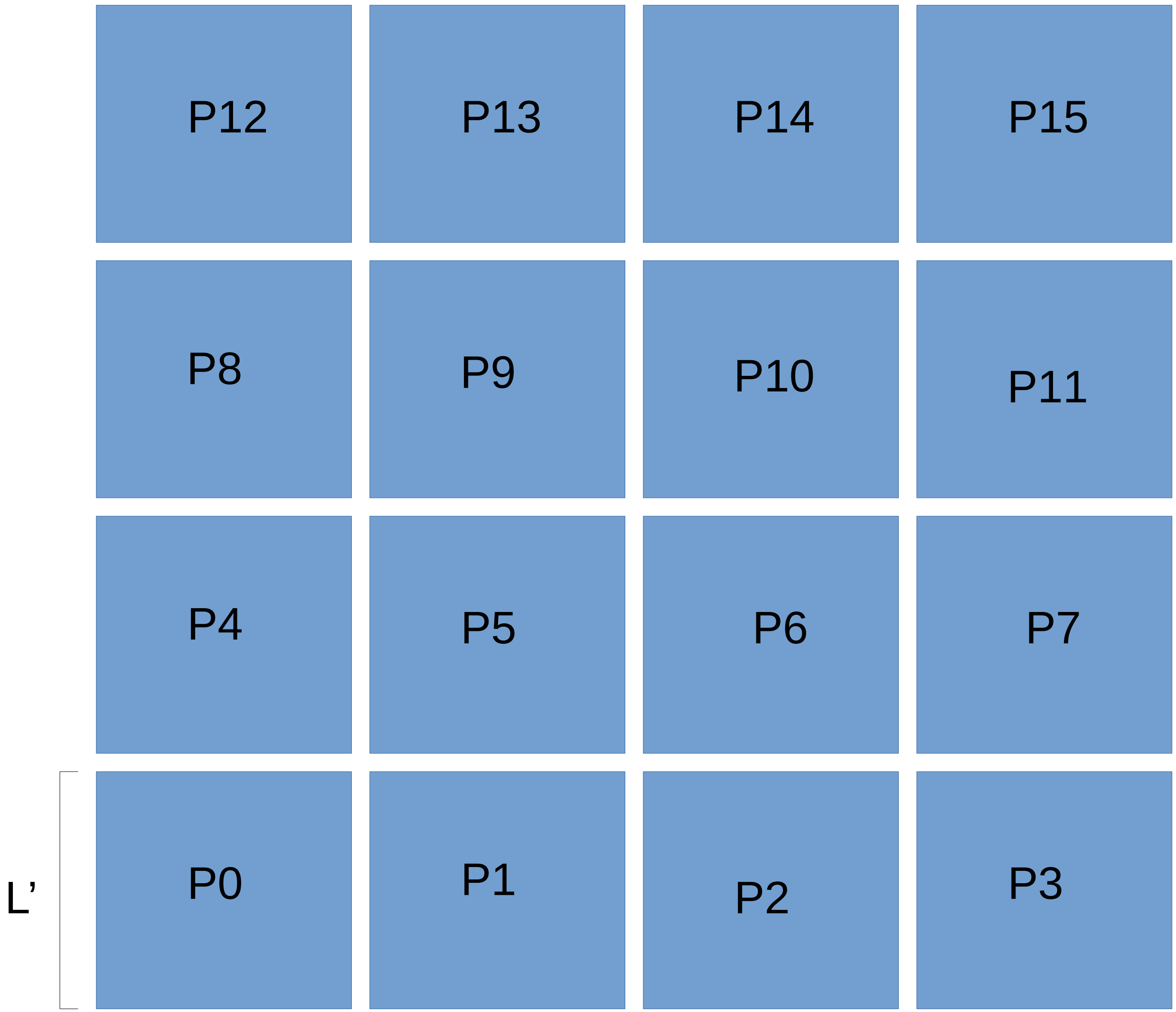
$$\left(\frac{\partial^2}{dx^2}+\frac{\partial^2}{dy^2}\right)u(x,y)= \ f\,(x,y) = 1.0\ inside\ \Omega$$

$$u(x,y) = 0\ on\ \partial\Omega$$

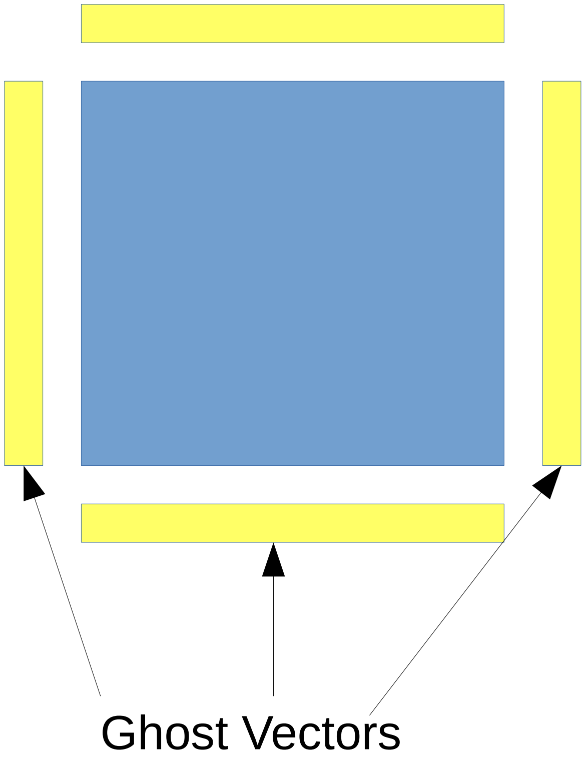


For 512x512 Mesh – Unit Square: Serial Code

Parallel Model



Like Parallel Jacobi – Using #Procs = A power of 4

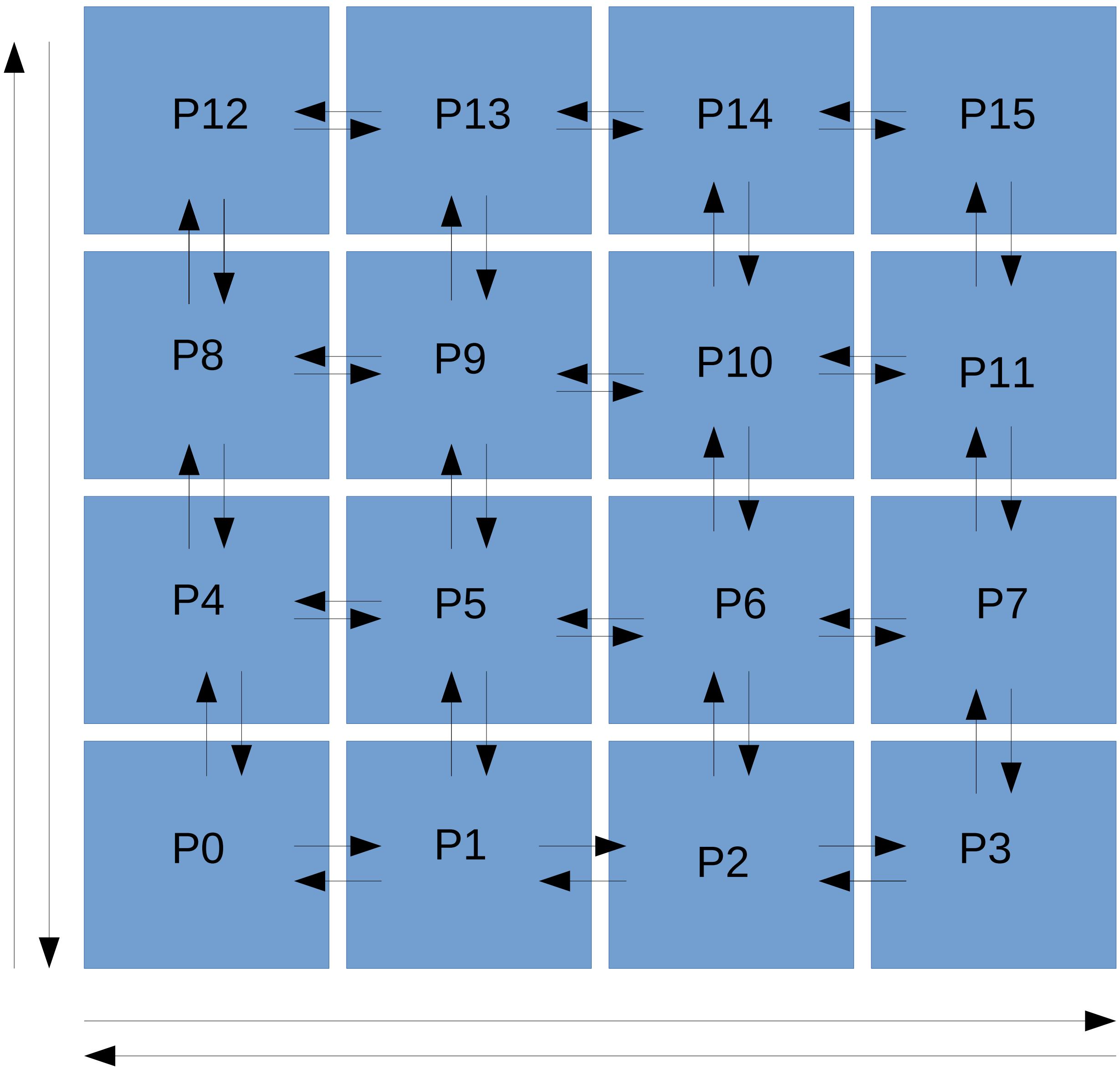


Procs Per Row/Col = Sqrt(Total Procs)

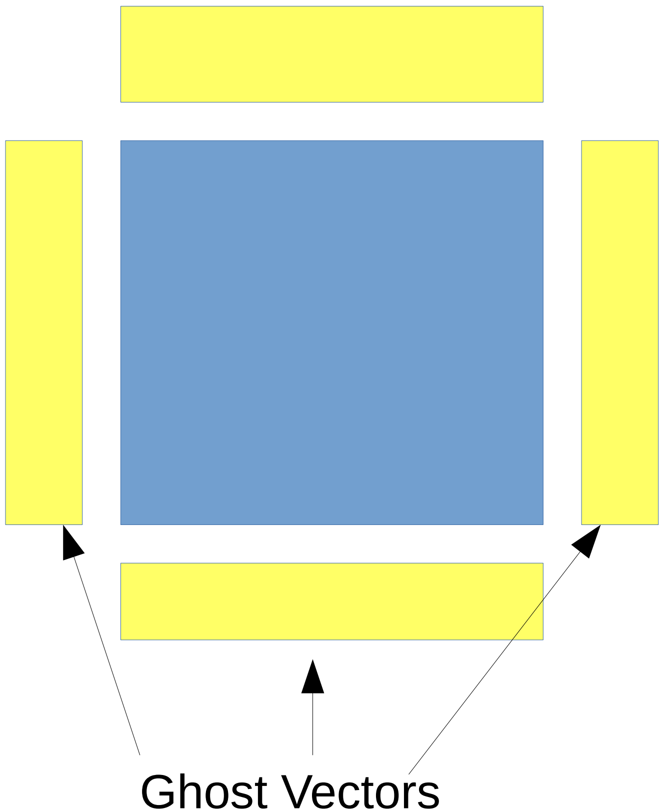
Divide Domain into P sub grids with Each Side’s L’ = N / sqrt(P)

Parallel Model

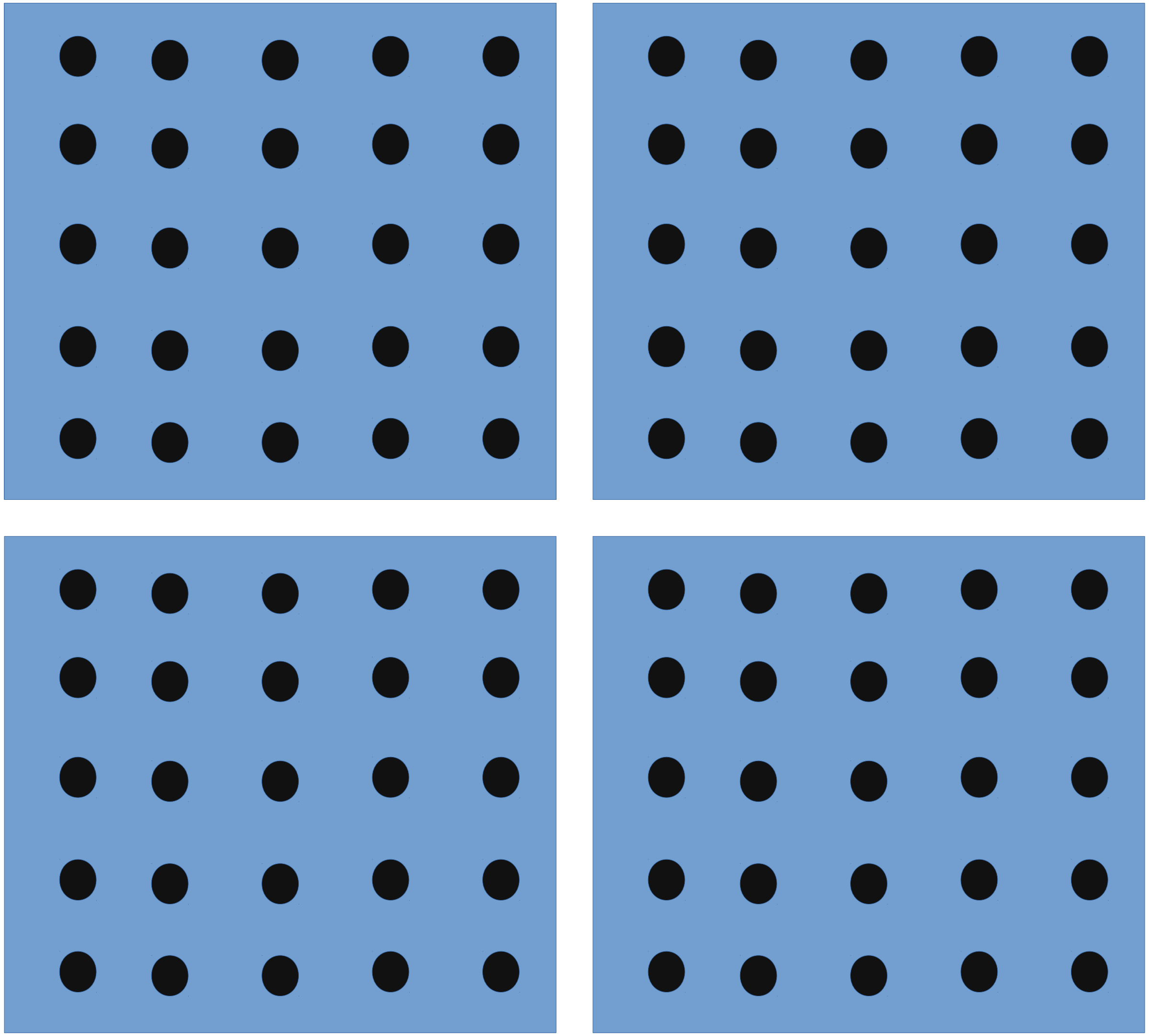
MPI Send/Recvs for Ghosts
1. Sweep Up/Down
2. Sweep Left/Right



Like Parallel Jacobi – Using
#Procs = A power of 4

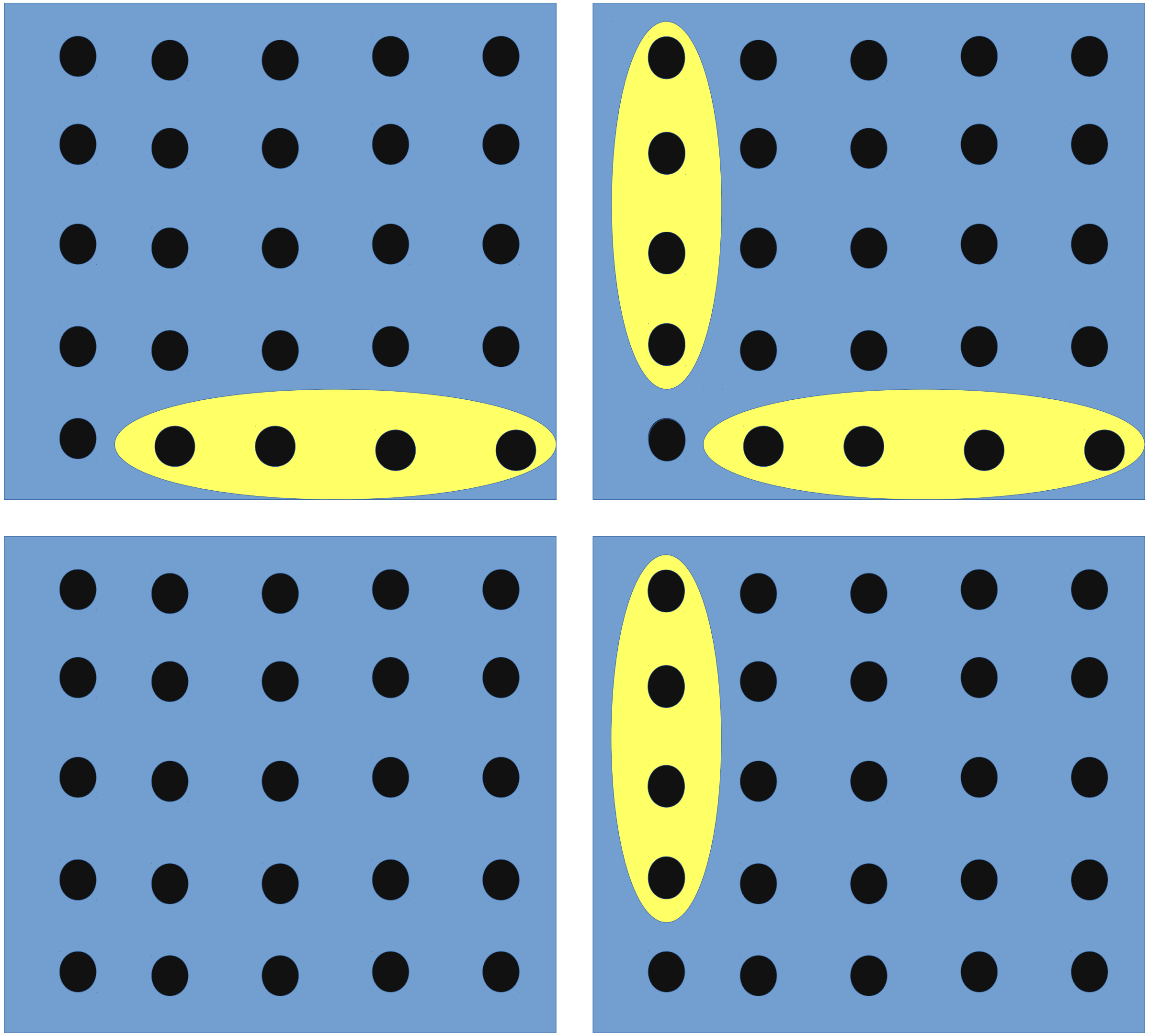


-9x9 Example (2^3 + 1) x (2^3 + 1)
-In memory its really a 10x10 Grid



Need a way to keep the Problem the Same
i.e. Number of Points per Dim = (2^m + 1) for
some m

-9x9 Example (2^3 + 1) x (2^3 + 1)
-In memory its really a 10x10 Grid

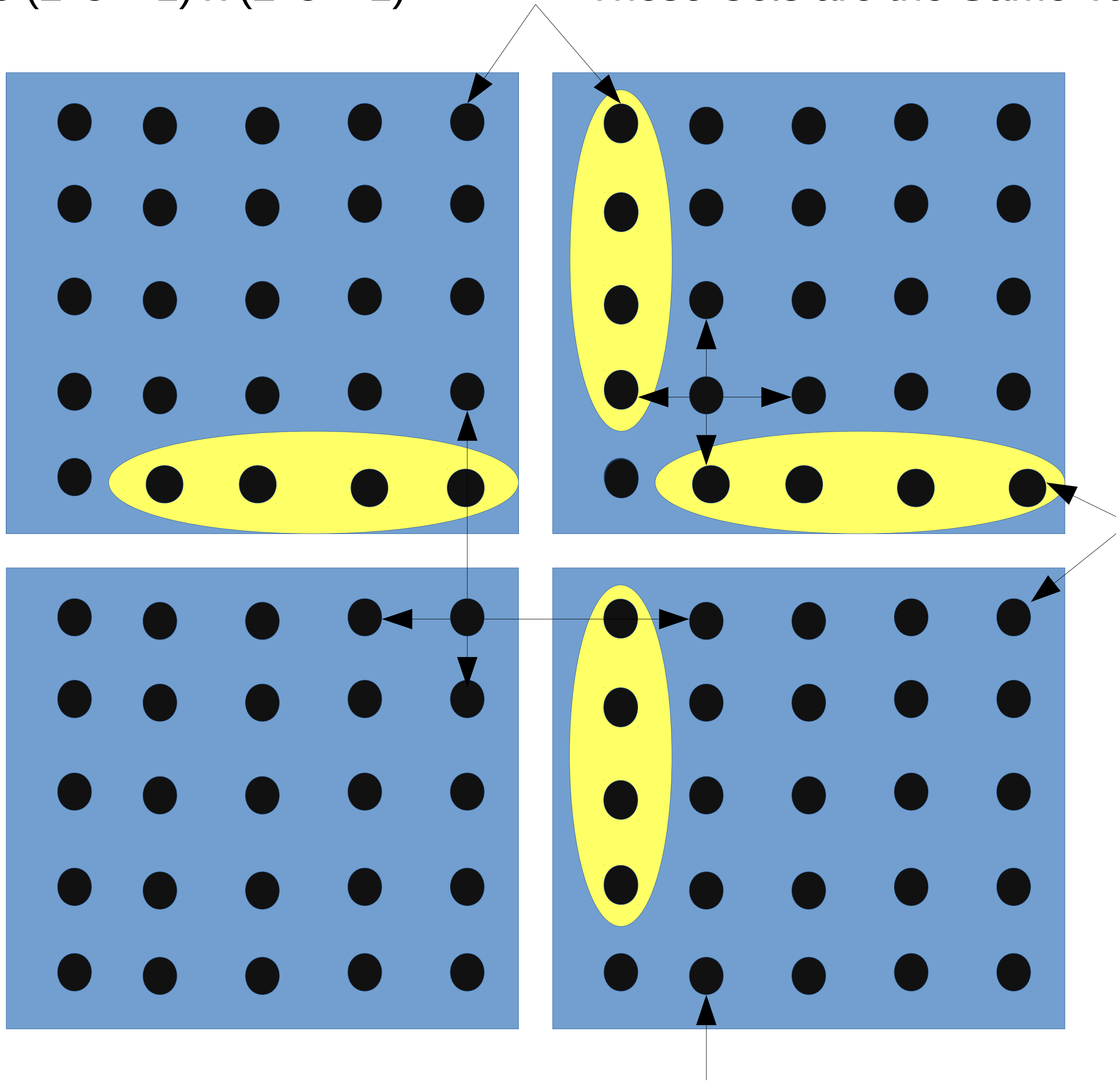


9x9 Example (2^3 + 1) x (2^3 + 1)

These Cols are the Same Values

-“Load” Left and Bottom Ghosts into 1st Column and Row (Unless BC’s apply) to Current Processor and relax normally On inner Points

-Relax Top row and Right Column using Values of top and right Ghosts (Unless BC’s apply!)



These Rows are the Same Values

It keeps the Problem the same for any number of MPI Tasks!

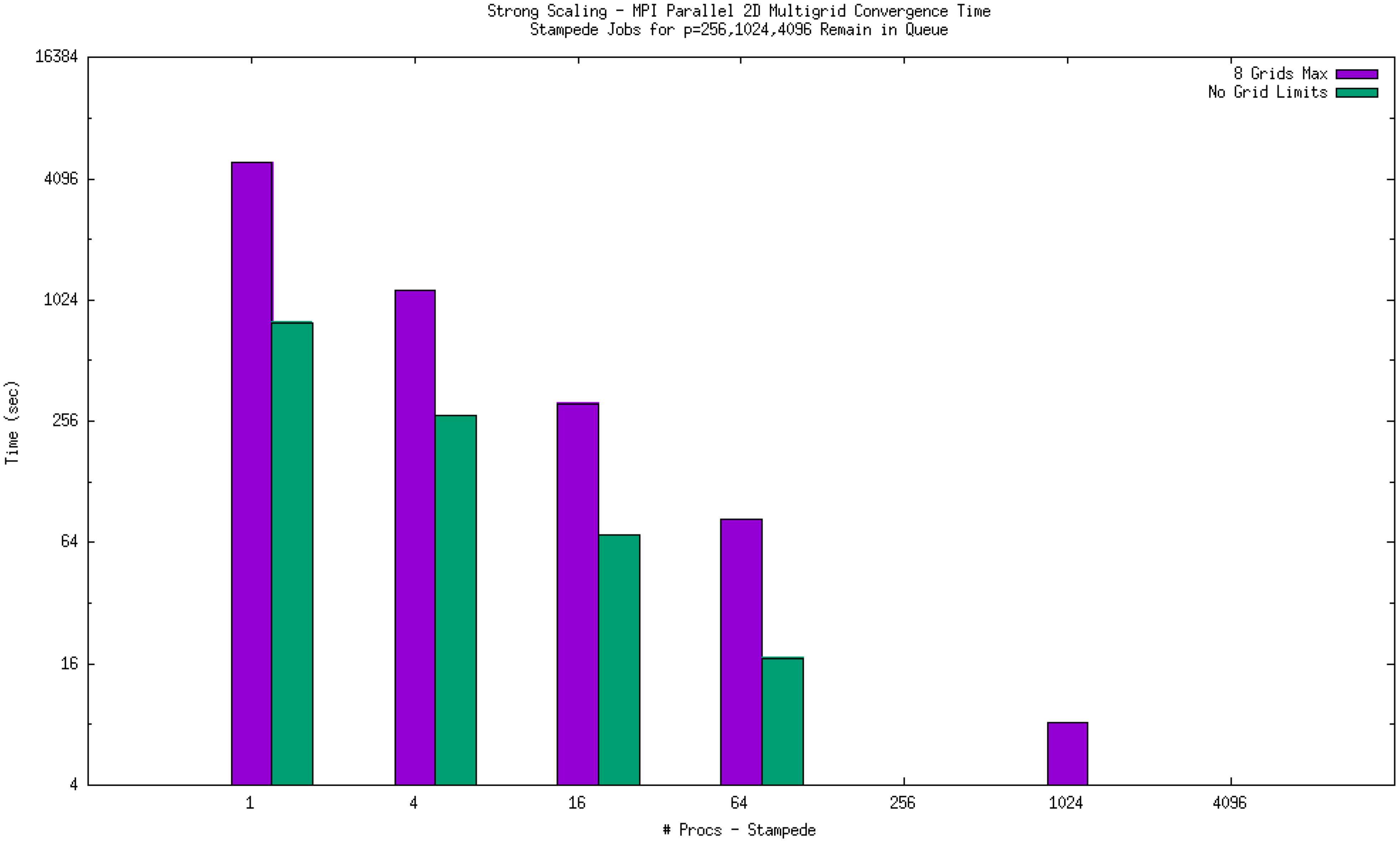
-Logically, this Column is the “Right Ghost” of the Bottom-Left Processor
-Same idea for Up/Down

Stampede Wait Times – Couldn't Reach Full Scale
https://portal.xsede.org/queue-prediction

STAMPEDE.TACC.XSEDE.ORG

QUEUE	WAIT TIME (HH:MM:SS)	START TIME
normal	253:23:50	Sun, 21 May 2017 09:57:31 GMT

Strong Scaling Results



Weak Scaling Results